



Calculating the Rotor Between Conformal Objects

J. Lasenby* , H. Hadfield  and A. Lasenby 

Abstract. In this paper we will address the problem of recovering covariant transformations between objects—specifically; lines, planes, circles, spheres and point pairs. Using the covariant language of conformal geometric algebra (CGA), we will derive such transformations in a very simple manner. In CGA, rotations, translations, dilations and inversions can be written as a single *rotor*, which is itself an element of the algebra. We will show that the rotor which takes a line to a line (or plane to a plane etc) can easily be formed and we will investigate the nature of the rotors formed in this way. If we can recover the rotor between one object and another of the same type, a useable metric which tells us how close one line (plane etc) is to another, can be a function of how close this rotor is to the identity. Using these ideas, we find that we can define metrics for a number of common problems, specifically recovering the transformation between sets of noisy objects.

1. Related Work

Our primary aim in this paper is to simultaneously estimate the *rotation* and *translation* that takes one object (line to line/circle to circle/plane to plane/sphere to sphere/point-pair to point-pair) to another. There are many methods that estimate rigid body transformations with points [1–4]. In [5] the authors estimate a general rotor between arbitrary objects using the idea of *carriers*—while interesting, this method lacks simplicity and does not deal directly with the objects themselves.

This article is part of the Topical Collection on Proceedings of AGACSE 2018, IMECC-UNICAMP, Campinas, Brazil, edited by Sebastià Xambó-Descamps and Carlile Lavor.

*Corresponding author.

2. Conformal Geometric Algebra

The objects we work with here will be CGA objects unless explicitly stated otherwise. We will use the standard extension of the 3D geometric algebra, where our 5D CGA space is made up of the standard spatial basis vectors $\{e_i\}$ $i = 1, 2, 3$, plus two additional basis vectors, e and \bar{e} with signatures, $e^2 = 1$, $\bar{e}^2 = -1$. Two *null vectors* can therefore be defined as: $n_\infty = e + \bar{e}$ and $n_0 = \frac{e - \bar{e}}{2}$. The mapping of a 3D vector x to its conformal representation X is given by $X = F(x) = \frac{1}{2}(x^2 n_\infty + 2x - 2n_0)$. Many of our target applications will be in computer vision, and in investigating algorithms which use more than just points, which is the case with most conventional computer vision algorithms.

3. A Rotor Between Objects

Suppose we wish to find the rotor (rotation, translation, dilation) which takes an object X_1 to an object X_2 (where X_1 and X_2 are conformal n -blades representing the lines/circles/planes/spheres/point pairs). If we firstly take lines as an example, conventionally we would translate along the common perpendicular and then rotate about the intersection point—which requires a series of non-trivial geometric operations for two arbitrary lines in space. Here we seek a method which will not require reverting to the geometric properties of the lines, but which will give the transformation in terms of the lines themselves—and we wish this method to be valid for all objects. In CGA, let the rotor which takes X_1 to X_2 be R_x , where this comprises both rotation, translation and dilation rotors. We assume both objects are normalised such that $X_1^2 = X_2^2 = \gamma$, where $\gamma = 1$ for lines, circles and point pairs, and $\gamma = -1$ for planes and spheres:

$$X_2 = R_x X_1 \tilde{R}_x$$

Note, that $\tilde{X} = -\gamma X$. We motivate our approach by considering the quantity $(X_1 + X_2)$ which is in some sense the ‘average’ object; ie, if we reflect X_1 in $(X_1 + X_2)$, we should get some function of X_2 (we assume for convenience that $X^2 = 1$, ie $\gamma = 1$):

$$\begin{aligned} (X_1 + X_2)X_1(X_1 + X_2) &= (1 + X_2X_1)(X_1 + X_2) \\ &= [2 + (X_1X_2 + X_2X_1)]X_2 \equiv KX_2 \end{aligned} \quad (1)$$

So the reflection does indeed produce a multiple, though the multiple is a *scalar plus 4-vector*, of X_2 . Since we can write the LHS of Eq. (1) as

$$\begin{aligned} (X_1 + X_2)X_1(X_1 + X_2) &= [(X_1 + X_2)X_1]X_1[X_1(X_1 + X_2)] \\ &= (1 + X_2X_1)X_1(1 + X_2X_1)^\sim \end{aligned}$$

we propose to use the spinor quantity $Z = 1 + \gamma X_2X_1$ to form R_x . As above (but now with γ included) $ZX_1\tilde{Z}$ gives;

$$Y = ZX_1\tilde{Z} = 2X_2 + \gamma(X_1X_2 + X_2X_1)X_2 = (2 + \gamma M_{12})X_2 = KX_2 \quad (2)$$

where $M_{12} = X_1X_2 + X_2X_1$ is the anticommutator of X_1 and X_2 . Thus, we see that Z takes X_1 to a multiple of X_2 , where this multiple involves the *anticommutator of the objects*. In general this anticommutator will have scalar and 4-vector parts (the bivector part of X_1X_2 cancels with the bivector part of X_2X_1).

Since all 4-vectors square to give a scalar, we can take $K^* = \langle K \rangle_0 - \langle K \rangle_4$, such that $KK^* = \langle K \rangle_0^2 - \langle K \rangle_4^2$, is a scalar, which we call μ . We show later that μ is always positive. We now multiply both sides of Eq. (2) by K^* to give:

$$\frac{1}{\mu} K^* Z X_1 \tilde{Z} = X_2$$

We now look to split up K^* such that $S^2 = K^*$, where $S = \alpha + \beta M_{12} \equiv (\alpha + \beta \langle M_{12} \rangle_0) + \beta \langle M_{12} \rangle_4$ and α and β are scalars. If S takes this form, it is clear that it is both self-reverse and commutes with Z and X_1 ; we can therefore write

$$\left(\frac{1}{\sqrt{\mu}} SZ \right) X_1 \left(\frac{1}{\sqrt{\mu}} SZ \right)^{\sim} = X_2$$

so that $\frac{1}{\sqrt{\mu}} SZ$ is our required rotor and $\mu = K^* K$. To find such an S we can use the square root formula given in [6] or simply equate scalar and 4-vector parts of the equation $S^2 = K^*$. We do the latter first in order to see how the particular form of our *scalar plus 4-vector* behaves and then confirm that it agrees with the formula in [6]:

$$(\alpha + \beta \langle M_{12} \rangle_0)^2 + 2\beta(\alpha + \beta \langle M_{12} \rangle_0) \langle M_{12} \rangle_4 + \beta^2 \langle M_{12} \rangle_4^2 = \langle K \rangle_0 - \langle K \rangle_4$$

Since $\langle K \rangle_0 = 2 + \gamma \langle M_{12} \rangle_0$ and $\langle K \rangle_4 = \gamma \langle M_{12} \rangle_4$, we have:

$$(\alpha + \beta \langle M_{12} \rangle_0)^2 + \beta^2 \langle M_{12} \rangle_4^2 = \langle K \rangle_0$$

$$2\beta(\alpha + \beta \langle M_{12} \rangle_0) \langle M_{12} \rangle_4 = -\gamma \langle M_{12} \rangle_4$$

From equating 4-vector parts we see that $2\beta(\alpha + \beta \langle M_{12} \rangle_0) = -\gamma$ so that, provided $\langle M_{12} \rangle_4 \neq 0$;

$$S = -\frac{\gamma}{2\beta} + \beta \langle M_{12} \rangle_4$$

If $\langle M_{12} \rangle_4 = 0$ we simply have $S = \sqrt{\langle K \rangle_0}$ if $\langle K \rangle_0$ is positive, which it is for lines, planes, circles and point pairs. $\langle K \rangle_0$ can take negative values for some sphere cases. If $\langle M_{12} \rangle_4 \neq 0$ we then find β from the equation which equates scalar parts:

$$\frac{1}{4\beta^2} - \beta^2 \lambda = \langle K \rangle_0$$

where $\langle M_{12} \rangle_4^2 \equiv \langle K \rangle_4^2 = -\lambda$, since the 4-vectors always square to give zero or a negative scalar. This is a quadratic in $u = \beta^2$:

$$4\lambda u^2 + 4\langle K \rangle_0 u - 1 = 0 \quad (3)$$

with solutions given by:

$$u = \frac{-4\langle K \rangle_0 \pm 4\sqrt{\langle K \rangle_0^2 + \lambda}}{8\lambda}$$

As $\beta^2 = u$ we need the solution which is guaranteed to be positive:

$$\beta^2 = \frac{1}{2\lambda} \left(\sqrt{\langle K \rangle_0^2 + \lambda} - \langle K \rangle_0 \right) = \frac{1}{2\lambda} (\sqrt{\mu} - \langle K \rangle_0)$$

Recall $K = 2 + \gamma(X_1 X_2 + X_2 X_1) = \langle K \rangle_0 + \langle K \rangle_4$, $K^* = \langle K \rangle_0 - \langle K \rangle_4$, $\lambda = -\langle K \rangle_4^2$, $\mu = K^* K = \langle K \rangle_0^2 + \lambda$, so is always positive (as $\lambda \geq 0$). We can now write the explicit form of the rotor as:

1. If $\langle M_{12} \rangle_4 \neq 0$:

$$R_x = \frac{1}{\sqrt{\mu}} \left(-\frac{1}{2\beta} + \beta \langle K \rangle_4 \right) (1 + \gamma X_2 X_1) \quad (4)$$

$$\beta^2 = \frac{1}{2(\sqrt{\mu} + \langle K \rangle_0)} \quad (5)$$

2. If $\langle M_{12} \rangle_4 = 0$ and $\langle K \rangle_0 > 0$

$$R_x = \frac{1}{\sqrt{\langle K \rangle_0}} (1 + \gamma X_2 X_1) \quad (6)$$

3. If $\langle M_{12} \rangle_4 = 0$ and $\langle K \rangle_0 < 0$,

$$R_x = \frac{1}{\sqrt{|\langle K \rangle_0|}} (1 + \gamma \bar{X}_2 X_1) \quad (7)$$

where $\bar{X}_2 = -X_2$ and $K' = 2 + \gamma(X_1 \bar{X}_2 + \bar{X}_2 X_1)$.

Taking the positive or negative square root for β simply changes the sign of the rotor, which makes no difference to the transformation. These expressions hold for all CGA objects: lines, planes, circles, spheres, point pairs. The following subsection will give the explicit forms for each of these objects and will discuss the third case which can occur for spheres.

Before looking in more detail at the nature of the rotors formed by the process outlined here, we return to Eq. (1) and note that we can now take X_1 to X_2 via a *reflection* in the quantity X_m where

$$X_m = \frac{S}{\sqrt{\mu}} (X_1 + X_2)$$

where S and μ are as given previously, ie $\mu = K^* K$ and S takes the form in Eqs. (4), (6) and (7) depending on the nature of M_{12} . We see in [7] that the quantity $\frac{S}{\sqrt{\mu}}$ projects the m -vector obtained from the addition of the two blades X_1 and X_2 onto an m -blade and therefore an object – the object being that in which we reflect X_1 in to get X_2 .

We can also confirm the solutions in Eqs. (4), (6) and (7) using the result in [6], where the square root of the *scalar plus 4-vector*, Σ , is given by

$$\sqrt{\Sigma} = \frac{\Sigma \pm [[\Sigma]]}{\sqrt{2\sqrt{\langle \Sigma \rangle \pm [[\Sigma]]}}} = \frac{\langle \Sigma \rangle \pm [[\Sigma]]}{\sqrt{2\sqrt{\langle \Sigma \rangle \pm [[\Sigma]]}}} + \frac{\langle \Sigma \rangle_4}{\sqrt{2\sqrt{\langle \Sigma \rangle \pm [[\Sigma]]}}}$$

where $[[\Sigma]] = \sqrt{\langle \Sigma \rangle^2 - \langle \Sigma \rangle_4^2}$. Here, our $\Sigma = K^* = \langle K \rangle_0 - \langle K \rangle_4$, so that (taking the solution corresponding to the + sign):

$$[[\Sigma]] = \sqrt{\mu} \quad \text{and} \quad \sqrt{2}\sqrt{\langle \Sigma \rangle + [[\Sigma]]} = \pm \frac{1}{\beta}$$

giving $\sqrt{\Sigma} = -\frac{1}{2\beta} + \beta\langle K \rangle_4$, as required (taking $-\frac{1}{\beta}$).

3.1. Lines

Conformal lines take the form $L = A \wedge B \wedge n$, with A, B being the conformal representations of two points lying on the line, and n the point at infinity. $\tilde{L} = -L$ and we normalise such that $L^2 = 1$, therefore $\gamma = 1$. For lines, the 4 vector part of the anticommutator takes the form $\beta I_5 n \equiv \beta I_3 n$, thus the square of this is always zero, which means $\lambda = 0$ and $\mu = \langle K \rangle_0^2$, which reduces Eq. (3) to $u = 1/(4\langle K \rangle_0)$ and $\beta = \pm 1/(2\sqrt{\langle K \rangle_0})$ [note that it does not matter which sign we take], giving us the simpler form of the rotor as:

$$R = \frac{1}{\langle K \rangle_0} \left(\sqrt{\langle K \rangle_0} - \frac{\langle K \rangle_4}{2\sqrt{\langle K \rangle_0}} \right) (1 + L_2 L_1) \equiv \frac{1}{\sqrt{\langle K \rangle_0}} \left(1 - \frac{\langle K \rangle_4}{2\langle K \rangle_0} \right) (1 + L_2 L_1) \quad (8)$$

3.2. Planes

With planes, as with lines, there is no issue of scaling as the objects are infinite. A plane Π is taken to be the conformal 4-blade of the form $A \wedge B \wedge C \wedge n$, with A, B, C any 3 conformal points lying on the plane. Conformal planes square to a negative number, so we assume that planes are normalised such that $\Pi^2 = -1$, therefore $\gamma = -1$. Note that $\tilde{\Pi} = \Pi$.

For planes the anticommutator is a scalar and it is not hard to show that (for normalised planes) $\langle K \rangle_0$ is always positive. Thus, the form for the rotor in the plane-to-plane case is particularly simple as the $\langle K \rangle_4$ term vanishes:

$$R_{\Pi} = \frac{1}{\sqrt{\langle K \rangle_0}} (1 - \Pi_2 \Pi_1) \quad (9)$$

where $K = 2 - (\Pi_1 \Pi_2 + \Pi_2 \Pi_1)$.

3.3. Circles

One might think that the case of circles-to-circles would be more complex, as a transformation which takes one arbitrary circle to another involves a dilation as well as a rotation and translation. However, nothing in the above derivation assumed anything specific about the rotor, and we find that we can use precisely the same formula to move between arbitrary circles.

Let us start with two conformal circles, C_1 and C_2 not necessarily of the same radius. A conformal circle is a 3-blade of the form $P \wedge Q \wedge R$, where P, Q, R lie on the circle. Circles square to a positive scalar, so we will assume that our circles are normalised such that $C^2 = 1$ and therefore $\gamma = 1$. Note that $\tilde{C} = -C$.

The anticommutator, M_{12} , is in general a *scalar plus 4-vector*, so we must use the form given in Eqs. (4) and (5) and little simplification is possible:

$$R_c = \frac{1}{\sqrt{\mu}} \left(-\frac{1}{2\beta} + \beta \langle K \rangle_4 \right) (1 + C_2 C_1) \quad (10)$$

$$\beta^2 = \frac{1}{2\lambda} (\sqrt{\mu} - \langle K \rangle_0) \quad (11)$$

with $K = 2 + (C_1 C_2 + C_2 C_1)$, $\mu = K^* K$.

3.4. Spheres

We start with two conformal spheres, S_1 and S_2 not necessarily of the same radius. A conformal sphere is a 4-blade of the form $N \wedge P \wedge Q \wedge R$, where N, P, Q, R lie on the sphere. Circles square to a negative scalar, so we will assume that our spheres are normalised such that $S^2 = -1$ and therefore $\gamma = -1$. Note that $\tilde{S} = S$.

As for planes, $\langle K \rangle_4^2$ is zero, so the rotor takes a very simple form:

$$R_s = \frac{1}{\sqrt{|\langle K \rangle_0|}} (1 - \tilde{S}_2 S_1) \quad (12)$$

where $K = 2 - (S_1 \tilde{S}_2 + \tilde{S}_2 S_1)$, $\tilde{S}_2 = S_2$ if $\langle K \rangle_0 > 0$ and $\tilde{S}_2 = -S_2$ if $\langle K \rangle_0 < 0$. $-S_2$ is the *same sphere* as S_2 , so in a sense it does not matter whether we take S_1 to S_2 or to $-S_2$ —this additional complexity occurs with spheres as they lack any intrinsic orientation, which is not the case for lines, planes, circles and point pairs.

3.5. Point Pairs

In the conformal setting, point pairs take the form $A \wedge B$ where A, B are conformal points — we can think of a point pair as a line segment. For a point pair, P , clearly $\tilde{P} = -P$ and P^2 gives a positive scalar. We will therefore assume that point pairs are normalised so that $P^2 = 1$.

Since the anticommutator will generally have both scalar and 4-vector parts, we again have the general form taken from Eqs. (4) and (5):

$$R_p = \frac{1}{\sqrt{\mu}} \left(-\frac{1}{2\beta} + \beta \langle K \rangle_4 \right) (1 + P_2 P_1) \quad (13)$$

$$\beta^2 = \frac{1}{2\lambda} (\sqrt{\mu} - \langle K \rangle_0) \quad (14)$$

with $K = 2 + (P_1 P_2 + P_2 P_1)$, $\mu = K^* K$.

3.6. Lines to Circles: Planes to Spheres

Note that in the previous rotor derivation we assumed X_1 and X_2 were blades of the same grade, but nothing further. Therefore, we should, and indeed do, find that the rotor formulae in Eqs. (4)–(7) work for moving between *lines and circles* and between *planes and spheres*.

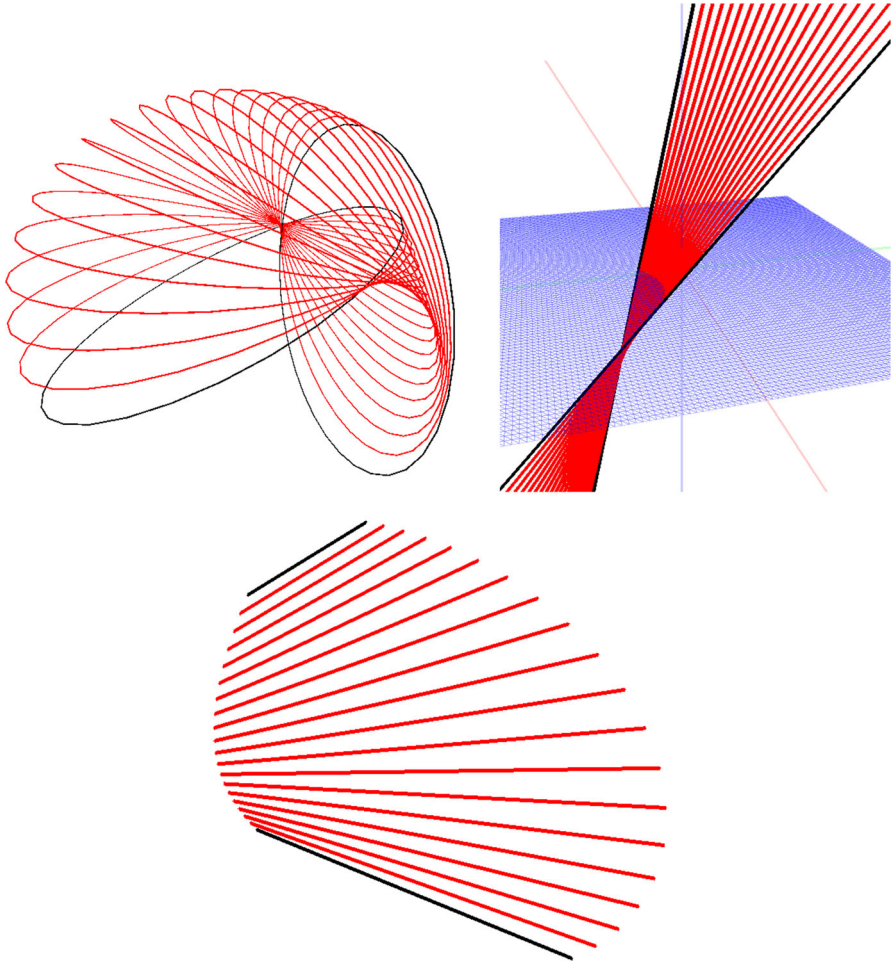


FIGURE 1. The red objects in each of these images show the interpolations of the rotors formed from pairs of black objects. Here we see that a range of geometric primitives including circles, lines and point pairs are all handled elegantly by the same framework

4. The Non-uniqueness of the Recovered Rotors

Although we have recovered rotors for each case of lines, planes, circles, spheres and point pairs, it is clear that these rotors are not unique. For example, if we transform one line into another, we can then translate along the second line without altering the result. So, a natural question to ask is *exactly what is the transformation we are recovering with the $1 + X_2X_1$ expression.*

To investigate this further we extract the bivector, B , for each recovered rotor, with $R = e^B$, and plot the interpolated objects for each of λ_i , $i =$

$1, \dots, n$, with $X_i = e^{\lambda_i B} X_1 e^{-\lambda_i B}$, where $X_2 = R X_1 \tilde{R}$ and $\lambda_i = i/n$. Figure 1 below show these interpolations for each class of object.

5. Conclusion

In this paper we have presented a general framework for explicitly extracting the conformal rotor that takes a conformal object of a given grade to another conformal object of the same grade. The technique works for point pairs, lines, circles, planes and spheres. In the process of investigating these rotors we have touched on the form of the object required to reflect one object into another and by visualising intermediate objects we have verified that the rotors take the objects smoothly to each other. Code that implements this rotor extraction algorithm is available in the clifford [8] python package and novel applications of this technique are additionally presented in [7, 9] and [10]. It is also interesting to note that the nature of the quantity $X_2 X_1$ was investigated first in [11], and then in [12], and noted to produce a quantity which was R^2 , where R is the rotor taking X_1 to X_2 . This has also been used for interpolations between objects in [13]. Here we have given explicit expressions for the rotor itself and investigated the whole range of use cases.

Open Access. This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

- [1] Eggert, D.W., Lorusso, A., Fisher, R.B.: Estimating 3-D rigid body transformations: a comparison of four major algorithms. *Mach. Vis. Appl.* **9**(5), 272–290 (1997)
- [2] Valkenburg, R., Dorst, L.: Estimating motors from a variety of geometric data in 3D conformal geometric algebra. *Guide to Geometric Algebra in Practice*, pp. 25–45 (2011)
- [3] Tingelstad, L., Egeland, O.: Motor estimation using heterogeneous sets of objects in conformal geometric algebra. *Adv. Appl. Clifford Algebras* **27**(3), 2035–2049 (2017)
- [4] De Keninck, S., Dorst, L.: Geometric algebra Levenberg–Marquardt. *Advances in Computer Graphics. CGI 2019. Lecture Notes in Computer Science* (2019)
- [5] Hitzer, E., Tachibana, K., Buchholz, S., Isseki, Y.: Carrier method for the general evaluation and control of pose, molecular conformation, tracking, and the like. *Adv. Appl. Clifford Algebras* **19**(2), 339–364 (2009)

- [6] Dorst, L., Valkenburg, R.: Square root and logarithm of rotors in 3D conformal geometric algebra using polar decomposition. *Guide to Geometric Algebra in Practice*, pp. 81–104 (2011)
- [7] Hadfield, H., Lasenby, J.: Direct linear interpolation of conformal geometric objects. *AACA: Topical Collection AGACSE 2018, IMECC-UNICAM, Campinas, Brazil* (2018)
- [8] Arsenovic, A., Hadfield, H., Kern, R.: The Pygae Team: pygae/clifford: v1.0.1 (2018)
- [9] Eide, E.R., Lasenby, J.: A novel way of estimating rotors between conformal objects and its applications in computer vision. *AACA: Topical Collection AGACSE 2018, IMECC-UNICAM, Campinas, Brazil* (2018)
- [10] Hadfield, H., Lasenby, J., Ramage, M., Doran, C.: Reform: rotor estimation from object resampling and matching. *AACA: Topical Collection AGACSE 2018, IMECC-UNICAM, Campinas, Brazil* (2018)
- [11] Lasenby, A.N., Lasenby, J. Wareham, R.J.: A covariant approach to geometry using geometric algebra. Technical Report, CUED/F-INFENG/TR-483, Cambridge University Engineering Department (2004)
- [12] Dorst, L., Fontijne, D., Mann, S.: Geometric algebra for computer science: an object-oriented approach to geometry. In: *Morgan Kaufmann series in computer graphics*. Elsevier, Amsterdam, Morgan Kaufmann, San Francisco (2007). ISBN: 978-0-12-369465-2
- [13] Colapinto, P.: *Spatial Computing with Conformal Geometric Algebra*. Phd thesis, University of California, Santa Barbara (2011)

J. Lasenby and H. Hadfield
Signal Processing and Communications Group, Department of Engineering
University of Cambridge
Cambridge
UK
e-mail: j1221@cam.ac.uk

H. Hadfield
e-mail: hh409@cam.ac.uk

A. Lasenby
Astrophysics Group, Cavendish Laboratory
University of Cambridge
Cambridge
UK
e-mail: a.n.lasenby@mrao.cam.ac.uk

Received: February 28, 2019.

Accepted: September 21, 2019.